Conditional Stroke Recovery for Fine-Grained Sketch-Based Image Retrieval

Anonymous ECCV submission

Paper id 6114

Abstract. The key to Fine-Grained Sketch Based Image Retrieval (FG-SBIR) is to establish fine-grained correspondence between sketches and images. Since sketches only consist of abstract strokes, stroke recognition ability plays an important role in FG-SBIR. However, existing works usually ignore the unique feature of sketches and treat images and sketches equally. Targeting at this problem, we propose Conditional Stroke Recovery (CSR) to enhance stroke recognition ability for FG-SBIR, in which we introduce an auxiliary task that requires the network recover the strokes using the paired image as condition. In this way, the network learns better to match the strokes with corresponding image elements. To complete the auxiliary task, we propose an unsupervised stroke disorder algorithm, which does well in stroke extraction and sketch augmentation. In addition, we figure out two weaknesses of the common triplet loss and propose double-anchor InfoNCE loss to reduce cosine distances between sketch-image pairs. Comprehensive experiments are conducted on four datasets (*i.e.*, QMUL-Shoe, QMUL-Chair, QMUL-ShoeV2, and Sketchy). In terms of acc@1, our method outperforms previous works by a great margin.

Keywords: Fine-Grained, Sketch-Based Image Retrieval, FG-SBIR

1 Introduction

Compared with text description, sketches interpret the information need of users more accurately in some real applications like product retrieval. As a result, the research of Sketch-Based Image Retrieval (SBIR) has received increasing attention recently. According to the retrieval granularity, SBIR can be categorized into Coarse-Grained category-level SBIR (CG-SBIR) [17, 4, 1] and Fine-Grained instance-level SBIR (FG-SBIR) [3, 37, 21, 43]. CG-SBIR retrieves an image in the gallery based on the category of query sketch while FG-SBIR retrieves a specific image that shares the same pose and outline. FG-SBIR has a wide range of applications in many fields such as searching online images or products. The key to FG-SBIR is to establish fine-grained correspondence between sketches and images. Considering that sketches only consists of strokes, to establish the fine-grained correspondence is mainly to match each sketch stroke with the corresponding image element (*e.g.*, a minute hand of a clock matched with an arrowed line in an sketch). We refer to the ability to recognize the arrowed line as

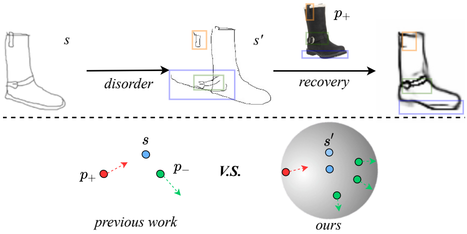

Fig. 1. 1) The top part shows an auxiliary recovery task conditioned by the paired image p_+. The task recovers a sketch s from a disordered one s'. The recovery process learns to match each stroke with its corresponding image element and place the stroke back to the right position. **2)** The bottom part compares triplet loss with our double-anchor InfoNCE loss. The former usually optimizes the L2 distance and samples negative images p_- inefficiently. The latter optimizes cosine similarity and take s_- as an augmented auxiliary anchor, therefore enhancing robustness of our model.

stroke recognition. However, the unique stroke characteristic is not well stressed in previous works [28, 30, 16].

Several previous works notice the problem but still remaining certain weaknesses. SaN [40, 41] feeds strokes of different drawing stages into the network to enhance stroke recognition ability. Whereas, pixel sequence annotations could be unavailable in real-world applications, limiting SaN inapplicable. A jigsaw pretraining strategy [22] was proposed to boost the ability of the network to capture fine-grained correspondence. However, the jigsaw game requires that the backgrounds of input images should be clean. A generative model [38] reconstruct sketch feature based on the paired image feature. In this way, the network interprets different image elements into sketch stroke precisely. However, uncertainties in manually created sketch make the method only available for coarse-grained SBIR.

To eliminate the uncertainties, we propose Conditional Stroke Recovery (CSR). We introduce an auxiliary task that recovers sketches. At first, we randomly move the strokes elsewhere within the sketch to obtain a corresponding disordered sketch. After that, conditioned on the image feature, we recover the original sketch from the feature of the disordered sketch. Intuitively, based on observation on the image, the network learns to decide which strokes are misplaced in the disordered sketch and place the misplaced strokes back. To decide the misplaced strokes is necessary for *stroke recovery*, so we take this intermediate recovery step as *misplaced stroke distinguishing* (MSD) for simplicity. In this way, the network learns to decide the right image element and match it with the corresponding sketch stroke, therefore enhancing stroke recognition ability. Besides, the disordering process is completed by our proposed unsupervised stroke disorder algorithm. Compared with previous works, CSR locates stroke recognition ability of the network to establish fine-grained correspondence between paired images and sketches. CSR does not require side information (*e.g.*, other pretraining datasets, stroke annotations and text descriptions), making it available in all kinds of scenarios. In addition, the existing works on FG-SBIR[32, 28, 30, 16] reduce the sketch-image domain gap mainly by triplet loss. However, triplet sampling might be inefficient and triplet loss might not support cosine distance retrieval well. Therefore, by accustoming InfoNCE loss[5] specially for FG-SBIR, we propose double-anchor InfoNCE loss to overcome the weaknesses of triplet loss. Comprehensive experiments on four benchmark datasets [28, 40] verify the effectiveness of CSR. Our main contributions are summarized as follows,

- We propose a novel method, Conditional Stroke Recovery (CSR), to learn stroke recognition ability.
- We introduce a double-anchor InfoNCE loss specially for FG-SBIR.
- We introduce a stroke disorder algorithm that can be applied to sketch augmentation without sequential information of pixels.
- Comprehensive results on four popular benchmark datasets demonstrate the advantage of our proposed CSR method over state-of-the-art methods.

2 Related Works

2.1 CG-SBIR

CG-SBIR is firstly proposed by [14]. CG-SBIR aims to learn a feature space that reduces the gap between sketch domain and image domain. Early CG-SBIR works usually extract edges maps from images and then design hand-crafted features to match the query sketches with images [8, 12, 11, 27]. In recent years, deep-learning based methods using variants of siamese losses [32] and ranking losses [42] are proposed for the CG-SBIR. Bui *et. al.* [4] applies a siamese loss and a triplet loss in different training stages to learn detailed features. A graph-based searching method[1] is proposed to re-rank the retrieved images.

2.2 FG-SBIR

FG-SBIR is firstly defined as retrieving the images with the same attributes (*e.g.*, viewpoint and body configuration) as the query sketch [15]. Qian *et. al.* [40] further extend the definition, requiring the retrieved image instance to be right corresponding to the query sketch. Since attribute annotations are usually unavailable in real-world applications, we follow the latter definition[40] like most of existing FG-SBIR works [36, 25, 16, 3, 21]. Apart from the huge domain gap in CG-SBIR, FG-SBIR also needs to capture fine-grained correspondence between images and sketches, making the task much more challenging. Li *et. al.* [15] adopt DPM [9] to learn different parts of objects and perform graph matching between sketches and images. Other existing FG-SBIR works [32, 28, 30, 16] are mostly based on a contrastive loss that captures fine-grained correspondence. Among these works, DSSA[32] proposes an attention mechanism to locate the most discrminative regions. GN Triplet [28] applies a classification loss to distinguish objects from different categories. Quadruplet [30] extends triplet loss to quadruplet loss for the same purpose. Radenovi *et. al.* [25] converts images to edge maps for shape matching. A reinforcement learning method [3] is proposed to support on-the-fly retrieval. TC-Net [16] further introduces various classification losses into the FG-SBIR task. Although SaN Triplet [41, 40] improve stroke recognition ability by incorporating strokes of different drawing phases into different input channels but it requires expensive stroke annotations. Besides, converting images to edge maps[41, 40, 22] for pre-training helps little for FG-SBIR[16] since edge maps can remove helpful information (*e.g.*, color and texture).

2.3 Sketch Reconstruction From Images

Bhunia *et. al.* [2] propose an image-to-sketch translation method. But it requires coordinate sequences and is only used for generation of more training sketch-image pairs. Pang *et. al.* [22] design a jigsaw game where edge maps are divided into grids and mixed into real image to form jigsaw tiles. However, simply dividing edge maps into grids might do great harm to semantic completeness of strokes. Yelamarthi *et. al.*[38] employ CVAE and CAAE to reconstruct sketch features from features of paired images. Since one image can correspond to more than one sketches, the generation process is not unique for the same image and therefore these generative frameworks do not work in FG-SBIR.

2.4 InfoNCE loss

Information Noise Contrastive Estimation loss (InfoNCE loss) is first proposed for Contrastive Predictive Coding [20], which learns robust representations by predicting the future in latent space. In SimCLR[5] and SimCLRv2[6], InfoNCE loss aimes to minimize the distance between paired augmented images. Chen *et. al.* [5] verify the advantage of InfoNCE loss over triplet loss on self-supervised image classification tasks. In this paper, based on InfoNCE loss, we introduce a double-anchor InfoNCE loss and prove its effectiveness on FG-SBIR as well.

3 Our Method

3.1 Overview

A fine-grained sketch-image dataset basically consists of paired sketch-image pairs. We denote a sketch by s and an image by p. For a given s, we represent the paired image by p_+, an unpaired one by p_-, and a disordered sketch by s'. Our proposed CSR mainly consists of two networks: a feature extraction network E^f and a stroke recovery network E^r, as illustrated in Fig. 2. E^f fuses multi-level features for retrieval. We give up the common triplet loss and propose double-anchor InfoNCE loss which supports cosine distance retrieval and samples negative images more efficiently. To learn stroke recognition ability, we design a recovery task by E^r. Given retrieval features of a disordered sketch and its paired image, E^r recovers the original sketch using a recovery loss.

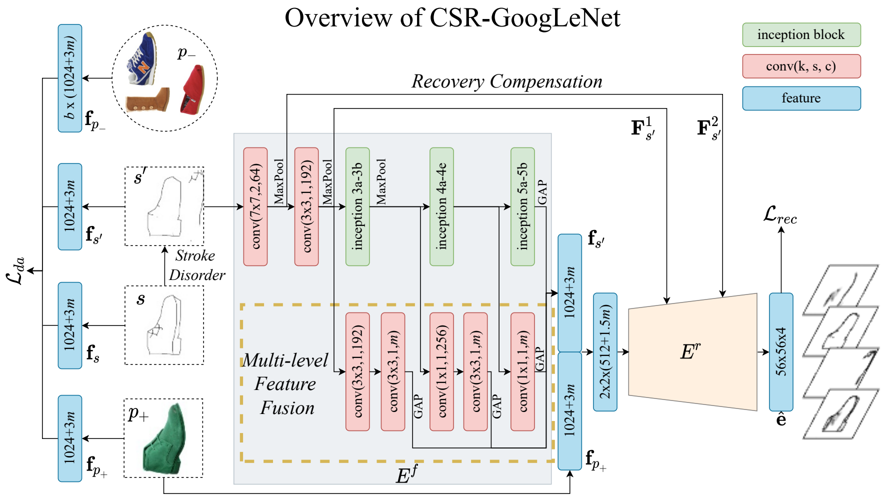

Fig. 2. Overview of CSR-GoogLeNet (refer to Supplementary for details). CSR consists of E^f and E^R. All of images, sketches and disordered sketches share the same E^f. In E^r, intermmediate feature maps are converted to m-dimensional feature vectors and then concatenated to the final 1024-dim output vector as a retrieval feature. An double-anchor InfoNCE loss $\mathcal{L}_{da}$ is applied on the retrieval feature. The retrieval features of a sketch s and its paired image p_+ are concatenated and sent into E^R. A recovery loss $\mathcal{L}_{rec}$ is applied on a 4-channel recovered extended map $\mathbf{e}$.

3.2 Stroke Disorder Algorithm

A disordered sketch s' is created by stroke disorder algorithm 1 in an unsupervised way. Algorithm 1 consists of two parts: *stroke extraction* (SE, Ln1-8) and *stroke disorder* (SD, Ln9-15). The whole algorithm runs as the following steps: **1)** Ln1 initializes the strokes as unconnected pixels; **2)** Ln3 obtains the longest stroke; **3)** Ln4 find a chokepoint where several strokes might join; **4)** Ln5-8 cut the stroke into n_m parts with a direct line. We want the line to be perpendicular to the variance of neighborhood pixels to avoid the line coinciding with the stroke. **5)** Ln9-15 disorders the selected strokes. The pixels removed in Ln6 into any stroke are stroke end points. We do not reassign them to any other strokes to prevent the network from taking a shortcut for stroke recovery[19]. The algorithm results are visualized in Fig. 3. p_d is a hyper-parameter and a larger p_d makes s' messier. Instead of simply dividing strokes into grids[22], the algorithm tries to preserve stroke completeness and independence. The SD process can be used as a sketch augmentation technique. The obtained sketch strokes can also be used for other augmentation approaches, (*e.g.*, stroke removal and stroke deformation in SaN[40, 29]).

Algorithm 1: Stroke disorder algorithm.

```
   Input: A $w \times h$ binary matrix $s$, where 0/1 is stroke/background color; A
          disorder probability $p_d$. The target strokes $n_s$, an integer.
   Output: Strokes of the disordered sketch.
 1 Group 0-valued pixels in $s$ into $n$ sets, say $\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_n$, where only pixels in
    the same set are connected, $\mathcal{S} \leftarrow \{\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_n\}$ ; //$\mathcal{P}_i$ forms a stroke
 2 while $n < n_s$ do // Stroke extraction
 3     $\mathcal{P}_m \leftarrow argmax_{\mathcal{P}_i \in \mathcal{S}} |\mathcal{P}_i|$ ;
 4     $c_m \leftarrow argmax_{c_i \in \mathcal{P}_m} |Adj(c_i) \cap \mathcal{P}_m|$; //$Adj$ yields neighborhood pixels.
 5     $\mathbf{v} \leftarrow var(Adj(c_m)), \mathcal{P}_l \leftarrow Ln(c_m, \frac{\mathbf{v}}{|\mathbf{v}|})$; // $var$ yields variance and
        $Ln(point, normal)$ yields a parameterized line
 6     $\mathcal{P}_m \leftarrow \mathcal{P}_m - \mathcal{P}_l$ ; // Remove intersection pixels
 7     Group pixels in $\mathcal{P}_m$ as Line-1 and obtain $n_m$ sets, say $\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_{n_m}$;
 8     $\mathcal{S} \leftarrow \mathcal{S} \cup \{\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_{n_m}\} - \{\mathcal{P}_m\}$, $n \leftarrow n - 1 + n_m$ ;
 9 end
10 Randomly select a subset $\mathcal{S}' \subset \mathcal{S}$, where $|\mathcal{S}'| = \lceil n \times p_d \rceil$;
11 for $\mathcal{P} \in \mathcal{S}'$ do // Stroke Disorder
12     $x_t \sim N(0, (w \times p_d)^2), y_t \sim N(0, (h \times p_d)^2), r \sim N(0, \pi^2 \times p_d^4)$;
13     Clip $x_t, y_t, r$ to constrain processed pixel coordinates in a valid value scope;
14     Rotate $\mathcal{P}$ by $r$ and translate $\mathcal{P}$ by $(x_t, y_t)$;
15 end
16 return $\mathcal{S}$
```

3.3 Feature Extraction

All of s, p_+, p_- and s' shares the same feature extraction procedure. We take s for example. Original output of E^f is a 1024-dimensional vector, mainly containing high-level information (*e.g.*, semantic information). We fuse multi-level features to enrich the output with low-level information (*e.g.*, contours, texture and color). We refer to this step as *Multi-level Feature Fusion*. Specifically, CNNs are utilized to encode intermmediate feature maps into m-dimensional vectors. They are fused to be a vector of $1024 + 3m$ dimensions, $\mathbf{f}_s$, which is taken as the final output of E^f. The procedure is formulated as $\mathbf{f}_s = E^f(\mathbf{x}_s)$. $\mathbf{f}_s$ is used for retrieval, so the distances between $\mathbf{f}_s$ and $\mathbf{f}_{p_+}$ should be reduced.

A common approach: triplet loss. A common approach is to employ a triplet loss $\mathcal{L}_{trp}$ on the retrieval feature:

$$\mathcal{L}_{trp} = max(d(\mathbf{f}_s, \mathbf{f}_{p_-}) - d(\mathbf{f}_s, \mathbf{f}_{p_+}) + r, 0), \tag{1}$$

where d is a distance metric and r is a margin. Empirically, $\mathcal{L}_{trp}$ should satisfy two important rules: **1)** the triplet anchor should be a sketch for sketch is the query domain. **2)** negative samples of $\mathcal{L}_{trp}$ should all be images. These two rules are adopted by most previous works[16, 28, 40, 30, 22]. Despite broad application of $\mathcal{L}_{trp}$, there exists two problems: **1)** Naive triplet sampling is inefficient: a batch of $b \times 3$ samples contains only b triplets. **2)** $\mathcal{L}_{trp}$ usually targets at L2 distance[16, 30, 32, 22] while cosine distance might produce better retrieval performance.

Most of existing works[16, 28, 40, 30, 22, 25] adopt L2 distance as d for $\mathcal{L}_{trp}$. Cosine distance disagrees with d because **(1)** cosine distance in $\mathcal{L}_{trp}$ leads to insignificant gradients; **(2)** cosine distance has a certain value scope and thus makes $\mathcal{L}_{trp}$ very sensitive to r. However, cosine distance performs better in retrieval since it avoids variations introduced by the vector norm [18]. Consequently, the inconsistency between training distance and retrieval distance might cause a performance bottleneck.

Our solution: double-anchor InfoNCE. Inspired by InfoNCE loss[5], we propose a loss that overcomes weaknesses of $\mathcal{L}_{trp}$ while following the above rules:

$$\mathcal{L}_{sa} = -\log \frac{e^{sim(\mathbf{f}_s, \mathbf{f}_{p_+})}}{e^{sim(\mathbf{f}_s, \mathbf{f}_{p_+})} + \sum_{p_-} e^{sim(\mathbf{f}_s, \mathbf{f}_{p_-})}}, \tag{2}$$

where sim yields amplified cosine similarity: $sim(x, y) = \frac{cos(x,y)}{\tau}$. τ controls temperature. Different from the original InfoNCE loss that targets at similarity of single-domain features, $\mathcal{L}_{sa}$ refers to the success of triplet loss and is designed to optimize cross-modality distances.

To further improve robustness of our model, the distance between s' and p_+ should be taken into consideration. Although s' is not strictly aligned with p_+, s' is more similar to p_+ than to p_-. At the same time, s should always be the closest to p_+. We therefore propose a double-anchor InfoNCE loss:

$$\mathcal{L}_{da} = -\log \frac{e^{sim(\mathbf{f}_s, \mathbf{f}_{p_+})} + \alpha e^{sim(\mathbf{f}_{s'}, \mathbf{f}_{p_+})}}{e^{sim(\mathbf{f}_s, \mathbf{f}_{p_+})} + \alpha e^{sim(\mathbf{f}_{s'}, \mathbf{f}_{p_+})} + \sum_{p_-} (e^{sim(\mathbf{f}_s, \mathbf{f}_{p_-})} + \alpha e^{sim(\mathbf{f}_{s'}, \mathbf{f}_{p_-})})}, \tag{3}$$

where $0 \leq \alpha < 1$. In $\mathcal{L}_{da}$, s' is an auxiliary anchor and regarded as an augmented sketch from s. A larger α encourages $\mathbf{f}_{p_+}$ to move towards $\mathbf{f}_{s'}$. To collect a b-sized training batch, we fill it with b triplets of (s, s', p_+). For a specific s or s', other $b - 1$ images in the same batch are regarded as as p_- . Although $\mathcal{L}_{trp}$ can also adopt a similar *efficient triplet collection* (ETC) approach, where $b \times (b - 1)$ triplets can be collected in a batch, $\mathcal{L}_{da}$ is still superior to $\mathcal{L}_{trp}$ since it directly optimizes cosine distances.

3.4 Stroke Recovery

The stroke recovery network E^r takes the input of $\mathbf{f}_{s'}$ and $\mathbf{f}_{p_+}$. The aim of E^r is to recover the paired sketch from its disordered style. To stabilize the training of E^r, we add another three maps to compose a 4-channel extended recovery map $\mathbf{e}$ as the recovery target, as illustrated in Fig. 3. Intuitively, the network first learns to distinguish misplaced and well-placed strokes (channel 2 and channel 3). After that, misplaced strokes are moved to their right places (channel 4) and then the sketch is recovered (channel 1).

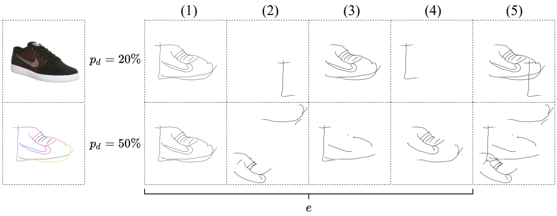

Fig. 3. Visualization of the stroke disorder algorithm 1. The leftmost two subfigures are an image and visualization of different groups of its paired sketch. The 5 columns on the right are: (1) the original sketch s; (2) disorder strokes, *i.e.*, $\mathcal{P} \in \mathcal{S}'$ after rotation and translation; (3) fixed strokes, *i.e.*, the unselected strokes $\mathcal{P} \in \mathcal{S} - \mathcal{S}'$; (4) selected strokes, *i.e.*, $\mathcal{P} \in \mathcal{S}'$ before rotation and translation; (5) the disorder sketch s'. We use column (1)-(4) to compose a 4-channel extended recovery map $\mathbf{e}$.

CSR does not intend to recover high-resolution sketch, so we downsample $\mathbf{e}$ to 56×56. It simplifies E^r and prevent the risk that $\mathbf{f}_{s'}$ contains over-detailed pixel-level information. We compensate $\mathbf{f}_{s'}$ and $\mathbf{f}_{p_+}$ by two low-level feature maps of s from E^f [26], denoted by $\mathbf{F}^1_{s'}$ and $\mathbf{F}^2_{s'}$. We refer to the trick as *recovery compensation*. The whole recovery process is formulated as: $\hat{\mathbf{e}} = E^r(\mathbf{f}_{s'}, \mathbf{f}_{p_+}, \mathbf{F}^1_{s'}, \mathbf{F}^2_{s'})$. Then we derive a recovery loss on the output $\hat{\mathbf{e}}$: $\mathcal{L}_{rec} = CE(\hat{\mathbf{e}}, \mathbf{e})$, where CE is the cross entropy loss. Although additional losses (*e.g.*, adversarial losses) might improve the quality of the recovered sketch, these losses turn out to be helpless for FG-SBIR when complicating our proposed framework.

3.5 Optimization

After obtaining $\mathcal{L}_{da}$ and $\mathcal{L}_{rec}$, we derive the total loss:

$$\mathcal{L}_{total} = w \times \mathcal{L}_{da} + \mathcal{L}_{rec}, \tag{4}$$

where w is the weight to balance $\mathcal{L}_{da}$ and $\mathcal{L}_{rec}$. All of s, p_+, p_- and s' share the same E^f. The parameters of both of E^f and E^r are optimized simultaneously.

4 Experiments

4.1 Experimental Setup

We briefly introduce the experimental setup in this section. Details can be seen in Supplementary. We report main results on GoogLeNet [33], InceptionV3 [34], DenseNet169 [13] and ResNet18 [10], which are adopted by the FG-SBIR works in the last three years [16, 2, 3, 22, 37, 7]. We conduct all other experiments based on GoogLeNet. Empirically, we set $m = 64, b = 96, w = 10.0, \tau = 0.005, n_s = 10$. p_d starts at 0.1 and linearly grows to 0.3. We dynamically set $\alpha = 1 - \alpha_p p_d, \alpha_p = 2$. We evaluate CSR on four fine-grained sketch-image datasets: Sketchy [28], QMUL-Shoe [40], QMUL-Chair [40], and QMUL-ShoeV2 [40]. Following Qian *et. al.* [40], we report top-1 accuracy (acc@1) and top-10 accuracy (acc@10).

Table 1. Comparison of CSR and baselines on Sketchy, QMUL-Chair, QMUL-Shoe and QMUL-ShoeV2. Column "Side" shows extra information used by existing works. "E" and "D" are short for extra annotations (*e.g.*, text descriptions and attributes) and other datasets [39, 24, 35, 22, 23]. Results are grouped by backbone. The best within each group are denoted in **boldface** and the best across different groups are **underlined**. "-": the result is missing in the original paper.

Method	Backbone	Side	Sketchy(%)		QMUL-Chair(%)		QMUL-Shoe(%)		QMUL-ShoeV2(%)	
			acc@1	acc@10	acc@1	acc@10	acc@1	acc@10	acc@1	acc@10
Song *et. al.* [31](BMVC'16)	SaN	E	-	-	78.4	99.0	50.4	91.3	-	-
SaN Triplet [40](CVPR'16)	SaN	DE	25.9	-	69.1	**97.9**	39.1	87.8	30.9	-
DSSA [32](ICCV'17)	SaN	E	-	-	81.4	95.9	**61.7**	**94.8**	**33.7**	-
Radenovi *et. al.* [25](ECCV'18)	VGG16	D	-	-	**85.6**	**97.9**	54.8	92.2	-	-
GN Triplet [28](TOG'16)	GoogLeNet	-	37.1	-	-	-	-	-	-	-
DCCRM [37](PR'19)	GoogLeNet	DE	46.2	**96.5**	-	-	-	-	-	-
Pang *et. al.* [22](CVPR'20)	GoogLeNet	D	-	-	86.0	-	56.5	-	36.5	-
CSR (ours)	GoogLeNet	-	**50.8**	85.4	**93.8**	**99.0**	**58.3**	**90.5**	**48.7**	**84.8**
Ayan *et. al.* [3] (CVPR'20)	InceptionV3	E	-	-	-	-	-	-	-	79.6
Bhunia *et. al.* [2] (CVPR'21)	InceptionV3	D	-	-	-	-	-	-	39.1	87.5
CSR (ours)	InceptionV3	-	**58.9**	**89.7**	**94.8**	**100.0**	**64.4**	**91.3**	**52.1**	**87.9**
Quadruplet [30] (ACMMM'17)	ResNet18	-	42.2	-	-	-	-	-	-	-
AE-Net [7](PR'22)	ResNet18	-	46.0	-	-	-	-	-	-	-
CSR (ours)	ResNet18	-	**47.6**	**82.9**	**93.8**	**97.9**	**55.5**	**89.1**	**45.1**	**85.5**
TC-Net [16](ACMMM'19)	Densnet169	E	40.8	-	95.9	100.0	63.5	94.8	40.2	-
CSR (ours)	Densenet169	-	**56.2**	**88.6**	**97.9**	**100.0**	**67.8**	**97.4**	**47.6**	**85.0**

4.2 Comparison With Existing Works

We compare CSR with 12 baseline methods and report results in Table. 1.

acc@1 Comparison: CSR outperforms previous works within each backbone group respectively. Compared with TC-Net[16], CSR yields better performance gain on large datasets, (*i.e.*, Sketchy and QMUL-ShoeV2). For small datasets (*i.e.*, QMUL-Chair and QMUL-Shoe), it is not important to collect more negative samples, so advantage of $\mathcal{L}_{da}$ is not abvious and the performance gain is smaller. Compared with previous works, CSR-Densenet169 achieves the best acc@1 on all four datasets, outperforming previous works by 10.0% on Sketchy, 2.0% on QMUL-Chair, 4.3% on QMUL-Shoe and 7.4% on QMUL-ShoeV2.

acc@10 Comparison: CSR achieves the best performance except on Sketchy, where DCCRM[37] reports 96.5%. We suspect that extra information (*e.g.*, text descriptions in DCCRM[37]) might provide rich references for matching similar images. However, the extra information can be ambiguous and help little to target at the exact image paired with the query. Therefore, acc@1 of DCCRM[37] is lower than CSR. Beside Sketchy, CSR achieves 100% on QMUL-Shoe Chair and outperforms TC-Net[16] by 2.6% on QMUL-Shoe and Bhunia *et. al.* by 0.4% on QMUL-ShoeV2.

Side Information Comparison: As shown in column "Side", some works[22, 3, 2, 37] rely on extra information. These methods dug into extra information in one or two specific fine-grained sketch-image datasets. However, in some other scenarios where such information is unavailable, these methods might suffer from performance drop. On the contrary, since CSR relaxes the requirement for extra information, it certainly paves the way for further applications.

To observe how extra information can work on CSR, we conducted an extra experiment: we pretrain CSR-GoogLeNet with $\mathcal{L}_{da}$ on the pretraining dataset used by Pang *et. al.* [22]. acc@1 on QMUL-Shoe improves from 64.4% to 68.7% and acc@1 on QMUL-ShoeV2 improves from 48.7% to 51.0%. The result shows the performance gain from extra information.

Table 2. acc@1/acc@10 results on QMUL-ShoeV2 and Sketchy. MF stands for multi-level feature fusion. RC stands for recovery compensation. S0 reports results obtained from a pretrained backbone. "SE+SD": we adopt the stroke disorder approach as a sketch augmentation technique by setting $p_d = 0.05$. "GridD": we divide the sketch into grids and then rotate and translate the grids to obtain a disordered sketch.

setting	MF	RC	total loss	Sketchy(%)		QMUL-ShoeV2(%)	
				acc@1	acc@10	acc@1	acc@10
S0	×	-	×	0.7	3.6	0.6	9.6
S1	×	-	$\mathcal{L}_{sa}$	42.6	80.2	42.2	82.9
S2	×	-	$\mathcal{L}_{sa}$ w/ SE+SD	43.4	81.4	44.2	83.2
S3	×	-	$\mathcal{L}_{da}$	44.9	81.6	43.7	83.0
S4	√	-	$\mathcal{L}_{da}$	48.2	82.4	45.9	84.3
S5	×	×	$\mathcal{L}_{rec} + \mathcal{L}_{da}$	45.2	82.5	44.0	83.7
S6	×	√	$\mathcal{L}_{rec} + \mathcal{L}_{da}$	47.7	84.8	46.6	**85.3**
S7	√	×	$\mathcal{L}_{rec} + \mathcal{L}_{da}$	48.9	84.4	46.7	83.5
S8	√	√	$\mathcal{L}_{rec} + \mathcal{L}_{da}$ w/ GridD	41.9	80.8	43.5	81.7
S9	√	√	$\mathcal{L}_{rec} + \mathcal{L}_{da}$	**50.8**	**85.4**	**48.7**	84.8

4.3 Ablation Study

We conduct ablation experiments on both QMUL-ShoeV2 and Sketchy bacause **1)** QMUL-ShoeV2 is the largest clean dataset; **2)** Sketchy is the largest and most diversified among the introduced four FG-SBIR dataset. Experiments on these two datasets can be representative.

Module Ablation We report ablation study results in Table. 2. S2 outperforms S1 by 0.4/2.2 acc@1 performance gain on Sketchy/QMUL-ShoeV2, implying that our disorder algorithm work well for sketch augmentation. However, S2 is still inferior to S3, which proves that $\mathcal{L}_{da}$ can effectively make the advantage of s' to enhance robustness of our model. Comparison between S4 and S3 shows importance of MF in FG-SBIR. Moreover, MF contributes a greater performance gain when $\mathcal{L}_{rec}$ is introduced (S5 *v.s.* S7). It is also observed that $\mathcal{L}_{rec}$ does not work when ablating MF and RC, which produce important low-level information to support the recovery process (S5 *v.s.* S3). S8 shows that simply dividing the sketch into grids is not viable because it destroys semantic completeness of sketch strokes. S9 is the best setting, demonstrating the effectiveness of our method.

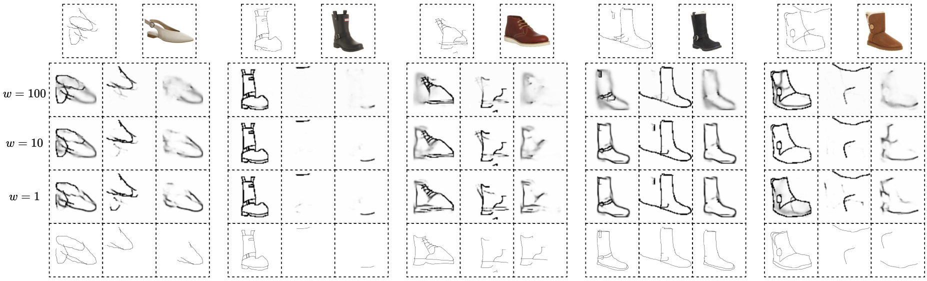

Fig. 4. Visualization of recovered sketches and retrieval features towards different w on QMUL-ShoeV2 test set. Within each group of figures bordered by dotted lines, the top two are the disordered sketch and the paired image. The rest three columns illustrate the original sketch, disordered strokes and selected strokes (refer to Fig. 3).

Effect of w We study effect of w in Fig. 4. The recovered sketches in QMUL-ShoeV2 using different w are visualized in Fig. 6. The results reveal that a smaller w yields clearer recovered sketches. An over small w will overemphasize $\mathcal{L}_{rec}$ so that retrieval feature contains too much pixel-level information. These information is unnecessary for retrieval. On the other hand, overvaluing w leads to performance drop. Based on the low quality of the recovered sketches with a large w, we suspect that $\mathcal{L}_{rec}$ is ignored during training so the model fails to learn stroke recognition ability.

Table 3. acc@1 results by ablating each channel of $\mathbf{e}$. "ablated" stands for the disenabled channel.

ablated	-	1	2	3	4
QMUL-ShoeV2	**48.7**	44.7	45.5	46.3	43.9
Sketchy	50.8	**51.0**	49.6	49.4	50.8

Table 4. acc@1 results by setting different α_p. Note that $\alpha = 1 - \alpha_p p_d$ and p_d linearly grows to 0.3.

α_p	0	1	2	3
QMUL-ShoeV2	38.8	48.3	**48.7**	47.3
Sketchy	36.7	50.3	**50.8**	49.6

Misplaced Stroke Distinguishing The visualization in Fig. 4 shows that recovery of disordered strokes (column 2) is better than that of selected strokes(column 3). Largely due to the complete pixel-level information, misplaced stroke distinguishing(MSD) is easier than stroke recovery. To further investigate the role MSD plays in stroke recovery in Table. 3, we ablate each channel of $\mathbf{e}$. That is, we remove the channel from $\mathbf{e}$ and study the retrieval performance. We report results in Tab. 3. On QMUL-ShoeV2, each channel shows its necessity since sketch-image pairs are well aligned and recovery is well performed. Whereas, in Sketchy, the image object can be occluded and the paired sketch can be misaligned. Then stroke recovery cannot complete but MSD is still achievable. Therefore, channel 2 abd channel 3 are more important than channel 1 and channel 4 on Sketchy and ablating channel 1 and channel 4 does not make much difference. In conclusion, when stroke recovery is unable to complete, we can resort to MSD to enhance stroke recognition ability.

Table 5. acc@1 results with variants of $\mathcal{L}_{trp}$ and $\mathcal{L}_{da}$. Column "loss" omits $\mathcal{L}_{rec}$. CSR adopts **A5** while A4 is the original InfoNCE loss[5]. "ETC" stands for efficient triplet collection (refer to Sec. 3.3). n_t is the number of triplets in a batch. "anchor" stands for the anchor modality.

setting	loss	anchor	ETC	n_t	*metric*	$b = 16$		$b = 96$	
						Sketchy	QMUL-ShoeV2	Sketchy	QMUL-ShoeV2
T1	$\mathcal{L}_{trp}$	s	×	b	L2	38.9	26.6	39.2	27.0
T2	$\mathcal{L}_{trp}$	p	×	b	L2	28.8	14.2	30.1	14.3
T3	$\mathcal{L}_{trp}$	$p\&s$	×	$2b$	L2	35.0	37.3	35.2	38.2
T4	$\mathcal{L}_{trp}$	s	$\surd$	$b(b-1)$	L2	45.8	43.0	46.5	45.2
T5	$\mathcal{L}_{trp}$	p	$\surd$	$b(b-1)$	L2	37.8	37.1	38.9	39.6
T6	$\mathcal{L}_{trp}$	$p\&s$	$\surd$	$4b(b-1)$	L2	45.2	42.0	46.2	44.5
T7	$\mathcal{L}_{trp}$	s	$\surd$	$b(b-1)$	cos	29.5	40.1	30.5	42.9
A1	$\mathcal{L}_{sa}$	s	-	-	cos	46.9	44.0	48.6	47.1
A2	$\mathcal{L}_{sa}$	s	-	-	dot	30.4	29.5	31.8	32.8
A3	$\mathcal{L}_{sa}$	p	-	-	cos	42.0	35.6	44.0	37.4
A4	$\mathcal{L}_{sa}$	$p\&s$	-	-	cos	46.8	37.5	48.7	39.0
A5	$\mathcal{L}_{da}$	$s\&s'$	-	-	cos	**48.3**	**44.7**	**50.8**	**48.7**
A6	$\mathcal{L}_{da}$	$s\&s'$	-	-	dot	30.0	30.1	32.7	32.8

$\mathcal{L}_{trp}$ *v.s.* $\mathcal{L}_{sa}$ *v.s.* $\mathcal{L}_{da}$ We compare variants of $\mathcal{L}_{trp}$, $\mathcal{L}_{da}$ and $\mathcal{L}_{da}$. Results are reported in Table. 5. Differences among some typical settings are visualized in Fig. 5. In this paper, we define as $\lambda(1 - cos(\cdot, \cdot))$, where λ is a scale hyper-parameter. Besides results in the Table. 5, we also conduct experiments of $\mathcal{L}_{trp} + dot$ and $\mathcal{L}_{da} + L2$ but their results are not comparable.

Results show that **1)** sketches serve as anchors better than images (T1 *v.s.* T2, T4 *v.s.* T5, A1 *v.s.* A3); **2)** negative samples should be all images and a sketch-image mixture does not help model performance (T4 *v.s.* T6, A1 *v.s.* A3) **3)** ETC is consistently superior to the naive triplet collection approach(T1-T3 *v.s.* T4-T6); **4)** $\mathcal{L}_{trp}$ is inconsistent with with cosine distance (T4 *v.s.* T7) ; **5)** $\mathcal{L}_{sa}/\mathcal{L}_{da}$ are inconsistent with dot similarity (A1 *v.s.* A3,A5 *v.s.* A6).

These results are reported according to retrieval with cosine distance. To ensure that the comparison is fair for $\mathcal{L}_{trp}$, we perform retrieval using L2 distance. In setting T4($b = 96$), the resultant acc@1 is 46.3/45.2 on Sketchy/QMUL-ShoeV2, which almost remains the same. We also compare $\mathcal{L}_{trp}$ acc@1-r curves using cosine and L2 distance in Fig.8. The curves show that $\mathcal{L}_{trp}+cos$ is sensitive to margin r and is not a feasible combination. These results verify that our $\mathcal{L}_{sa}$ and $\mathcal{L}_{da}$ are superior to $\mathcal{L}_{trp}$. Besides, enlarging the batch size leads to greater performance gain in T4-T6, A1 and A2-A4. This is because these settings can collect more negative samples.

Augmentation via Stroke Disorder To further investigate how our proposed stroke disorder algorithm works for sketch augmentation, we reproduce several existing worksGN Triplet[28] [28] and Quadruplet[30] and TC-Net[16] and employ different sketch augmentation methods in the training stage. The results are reported in Table. 6. In this section, SaN stands for the augmentation methods

Fig. 5. Illustration of some settings in Tab. 5.

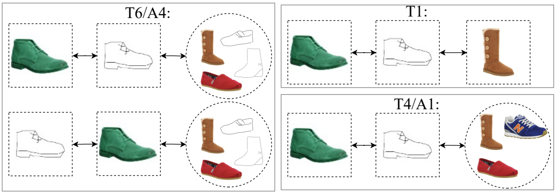

Table 6. Acc@1 using different augmentation methods on Sketchy.

augmentation	[28]	[30]	[16]	$\mathcal{L}_{sa}$
baselines	37.1	42.2	40.8	42.6
SE+SD	37.6	44.1	41.4	43.4
SE+SaN	39.0	43.9	41.7	43.6
SE+SD+SaN	39.3	44.4	42.2	44.0

applied in SaN[40], including stroke removal and stroke deformation. p_d is set to 0.05. Column "$\mathcal{L}_{sa}$" reports results of setting S2 in Tab. 2. Results show that **1)** SE+SD outperforms baselines, verifying that SD can improve diversity of training samples and prevent the risk of overfitting; **2)** both of SE+SaN and SE+SD+SaN outperform baselines, implying that extracted strokes are effective; **3)** SaN works as an augmentation method better than SD. Compared with rotation and translation of SD, removal and deformation can provide more natural augmented sketches. But we do not disorder a sketch via removal and deformation since sketches augmented in this way are found difficult for recovery.

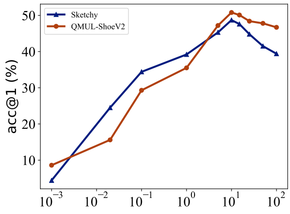

Fig. 6. acc@1-w curves of CSR.

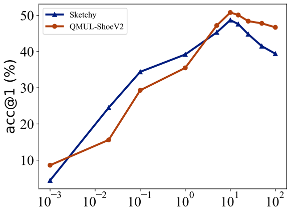

Fig. 7. acc@1-τ curves of CSR with different *sim*.

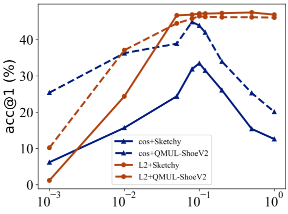

Fig. 8. acc@1-r curves using $\mathcal{L}_{trp}$ with different d.

Effect of $\boldsymbol{\tau}$ We plot acc@1-τ curves in Fig. 7. The results show that cosine similarity works better than dot product. An extremely small τ can lead good model performance, which is subtly different from observation in SimCLR[5]. SimCLR[5] aims at self-supervised learning task, where a pair of positive samples are augmented from the same image and their cosine similarity is very high, approximating one. In FG-SBIR, a sketch-image pair belongs to two different domains and their features are likely to be mutually orthogonal. Therefore, a small τ is required to amplify the similarity between a sketch-image pair.

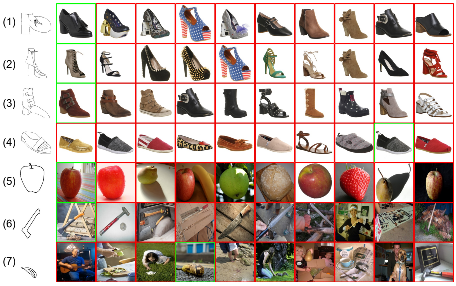

Fig. 9. Top-10 retrievals on QMUL-ShoeV2 (row (1)-(4)) and Sketchy (row (5)-(7)). Green/red borders indicate the correct/incorrect retrieved images.

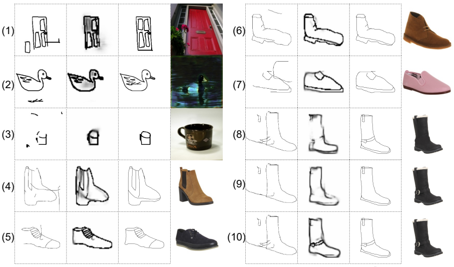

Fig. 10. Recovery visualization on Sketchy (row (1)-(3)) and QMUL-ShoeV2 (row (4)-(6)). The four columns are the disordered sketch, the recovered sketch, the original sketch and the paired image.

5 Case Study

Retrieval results are visualized in Fig. 9. In most cases, the first retrieved image is correctly hit. The failure case of row (4) results from misalignment between the sketch and target images. The failure case of row (7) is due to the noises from image background. Recovered sketches are presented in Fig. 10. Recovery results are generally satisfactory and most of poor-quality recovery results come from Sketchy. For example, the duck in row (2) is not well recovered for **1)** the feather in the sketch is abstract and does not match any image element; **2)** bottom of the duck is obstructed by water, making stroke recovery unfeasible. These two problems are common in Sketchy. Sketchy therefore benefits less from $\mathcal{L}_{rec}$ than QMUL-ShoeV2. Moreover, compared with row(10), we mannually erase the knot of the image in row (8) and the knot of the disordered sketch in row (9). In both circumstances of row (8) and row (9), the knot cannot be recovered. These three cases imply that the recovery process does not rely on memorizing category characteristics. Instead, the recovery needs to match each sketch stroke with the corresponding image element.

6 Conclusion

In this paper, we investigate fine-grained sketch-based image retrieval (FG-SBIR) from a new perspective. we highlight stroke recognition ability of the network and propose Conditional Stroke Recovery(CSR). CSR introduce an auxiliary task that recovers a sketch from its disordered style with its paired image as condition. Moreover, targeting at the weaknesses of triplet loss, we propose double-anchor InfoNCE loss specially for FG-SBIR. A unsupervised stroke disorder algorithm is also proposed along with CSR. The algorithm can work as a new sketch augmentation approach. In terms of acc@1, CSR outperforms previous works by a great margin on four datasets.

References

1. Bhattacharjee, S.D., Yuan, J., Hong, W., Ruan, X.: Query adaptive instance search using object sketches. In: Proceedings of the 2016 ACM Conference on Multimedia Conference, MM (ACM MM). pp. 1306–1315 (2016)
2. Bhunia, A.K., Chowdhury, P.N., Sain, A., Yang, Y., Xiang, T., Song, Y.: More photos are all you need: Semi-supervised learning for fine-grained sketch based image retrieval. In: Computer Vision and Pattern Recognition (CVPR) (2021)
3. Bhunia, A.K., Yang, Y., Hospedales, T.M., Xiang, T., Song, Y.Z.: Sketch less for more: On-the-fly fine-grained sketch-based image retrieval. In: Computer Vision and Pattern Recognition (CVPR) (2020)
4. Bui, T., Ribeiro, L.S.F., Ponti, M., Collomosse, J.P.: Sketching out the details: Sketch-based image retrieval using convolutional neural networks with multi-stage regression. Computers & Graphics (CAG) **71**, 77–87 (2018)
5. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.E.: A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning, (ICML) (2020)
6. Chen, T., Kornblith, S., Swersky, K., Norouzi, M., Hinton, G.E.: Big self-supervised models are strong semi-supervised learners. In: Advances in neural information processing systems (NIPS) (2020)
7. Chen, Y., Zhang, Z., Wang, Y., Zhang, Y., Feng, R., Zhang, T., Fan, W.: Ae-net: Fine-grained sketch-based image retrieval via attention-enhanced network. Pattern Recognition(PR) (2022)
8. Eitz, M., Hildebrand, K., Boubekeur, T., Alexa, M.: An evaluation of descriptors for large-scale image retrieval from sketched feature lines. Computers & Graphics **34**(5), 482–498 (2010)
9. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **32**(9), 1627–1645 (2010)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Computer Vision and Pattern Recognition (CVPR) (2016)
11. Hu, R., Collomosse, J.: A performance evaluation of gradient field hog descriptor for sketch based image retrieval. Computer Vision and Image Understanding (CVIU) (2013)
12. Hu, R., Wang, T., Collomosse, J.: A bag-of-regions approach to sketch-based image retrieval. In: IEEE International Conference on Image Processing (ICIP) (2011)
13. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Computer Vision and Pattern Recognition (CVPR). pp. 2261–2269 (2017)
14. Kato, T., Kurita, T., Otsu, N., Hirata, K.: A sketch retrieval method for full color image database-query by visual example. In: IAPR International Conference on Pattern Recognition (ICPR). pp. 530–533 (1992)
15. Li, Y., Hospedales, T.M., Song, Y.Z., Gong, S.: Intra-category sketch-based image retrieval by matching deformable part models. In: British Machine Vision Conference (BMVC). pp. 115.1–115.12 (2014)
16. Lin, H., Fu, Y., Lu, P., Gong, S., Xue, X., Jiang, Y.G.: Tc-net for isbir: Triplet classification network for instance-level sketch based image retrieval. In: ACM International Conference on Multimedia (ACM MM) (2019)
17. Liu, L., Shen, F., Shen, Y., Liu, X., Shao, L.: Deep sketch hashing: Fast free-hand sketch-based image retrieval. In: computer vision and pattern recognition (CVPR). pp. 2862–2871 (2017)

18. Liu, Q., Xie, L., Wang, H., Yuille, A.: Semantic-aware knowledge preservation for zero-shot sketch-based image retrieval. In: ICCV (2019)
19. Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: European Conference on Computer Vision (ECCV) (2016)
20. van den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. CoRR **abs/1807.03748** (2018)
21. Pang, K., Li, K., Yang, Y., Zhang, H., Hospedales, T.M., Xiang, T., Song, Y.Z.: Generalising fine-grained sketch-based image retrieval. In: Computer Vision and Pattern Recognition (CVPR). pp. 677–686 (2019)
22. Pang, K., Yang, Y., Hospedales, T.M., Xiang, T., Song, Y.: Solving mixed-modal jigsaw puzzle for fine-grained sketch-based image retrieval. In: Computer Vision and Pattern Recognition (CVPR) (2020)
23. Peng, C., Gao, X., Wang, N., Li, J.: Face recognition from multiple stylistic sketches: Scenarios, datasets, and evaluation. Pattern Recognit. **84**, 262–272 (2018)
24. Radenovic, F., Tolias, G., Chum, O.: CNN image retrieval learns from bow: Unsupervised fine-tuning with hard examples. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 3–20 (2016)
25. Radenovic, F., Tolias, G., Chum, O.: Deep shape matching. In: European Conference on Computer Vision (ECCV). pp. 751–767 (2018)
26. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., III, W.M.W., Frangi, A.F. (eds.) Medical Image Computing and Computer-Assisted Intervention MICCAI. vol. 9351, pp. 234–241 (2015)
27. Saavedra, J.M., Barrios, J.M., Orand, S.: Sketch based image retrieval using learned keyshapes (LKS). In: British Machine Vision Conference (BMVC). pp. 164.1–164.11 (2015)
28. Sangkloy, P., Burnell, N., Ham, C., Hays, J.: The sketchy database: Learning to retrieve badly drawn bunnies. ACM Transactions on Graphics (TOG) (2016)
29. Schaefer, S., McPhail, T., Warren, J.D.: Image deformation using moving least squares. ACM Trans. Graph. (2006)
30. Seddati, O., Dupont, S., Saïd, M.: Quadruplet networks for sketch-based image retrieval. In: ACM International Conference on Multimedia (ACM MM) (2017)
31. Song, J., Song, Y.Z., Xiang, T., Hospedales, T., Xiang, R.: Deep multi-task attribute-driven ranking for fine-grained sketch-based image retrieval. In: British Machine Vision Conference (BMVC). vol. 1, p. 3 (2016)
32. Song, J., Yu, Q., Song, Y.Z., Xiang, T., Hospedales, T.M.: Deep spatial-semantic attention for fine-grained sketch-based image retrieval. In: International Conference on Computer Vision (ICCV). pp. 5551–5560 (2017)
33. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: computer vision and pattern recognition (CVPR). pp. 1–9 (2015)
34. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Computer Vision and Pattern Recognition (CVPR). pp. 2818–2826 (June 2016)
35. Xu, P., Huang, Y., Yuan, T., Pang, K., Song, Y., Xiang, T., Hospedales, T.M., Ma, Z., Guo, J.: Sketchmate: Deep hashing for million-scale human sketch retrieval. In: Computer Vision and Pattern Recognition (CVPR). pp. 8090–8098 (2018)
36. Xu, P., Yin, Q., Qi, Y., Song, Y.Z., Ma, Z., Wang, L., Guo, J.: Instance-level coupled subspace learning for fine-grained sketch-based image retrieval. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 19–34 (2016)

37. Yanfei, W., Fei, H., Yuejie, Z., Rui, F., Tao, Z., Weiguo, F.: Deep cascaded cross-modal correlation learning for fine-grained sketch-based image retrieval. Pattern Recognition(PR) (2019)
38. Yelamarthi, S.K., Reddy, S.K., Mishra, A., Mittal, A.: A zero-shot framework for sketch based image retrieval. In: ECCV (2018)
39. Yu, A., Grauman, K.: Fine-grained visual comparisons with local learning. In: Computer Vision and Pattern Recognition (CVPR). pp. 192–199 (2014)
40. Yu, Q., Liu, F., Song, Y.Z., Xiang, T., Hospedales, T.M., Loy, C.C.: Sketch me that shoe. In: Computer Vision and Pattern Recognition (CVPR) (2016)
41. Yu, Q., Yang, Y., Liu, F., Song, Y.Z., Xiang, T., Hospedales, T.M.: Sketch-a-net: A deep neural network that beats humans. International journal of computer vision (IJCV) **122**(3), 411–425 (2017)
42. Zhu, H., Long, M., Wang, J., Cao, Y.: Deep hashing network for efficient similarity retrieval. In: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI). pp. 2415–2421 (2016)
43. Zhu, M., Chen, C., Wang, N., Tang, J., Bao, W.: Gradually focused fine-grained sketch-based image retrieval. PLoS ONE **14**(5), e0217168 (2019)